Mockito exercises : Exercise 1: Mocking and Stubbing

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

import static org.junit.jupiter.api.Assertions.\*;

// Simulated external API interface

interface ExternalApi {

String getData();

}

// Service class using the external API

class MyService {

private ExternalApi api;

public MyService(ExternalApi api) {

this.api = api;

}

public String fetchData() {

return api.getData();

}

}

public class MyServiceTest {

@Test

public void testExternalApi() {

ExternalApi mockApi = Mockito.mock(ExternalApi.class);

when(mockApi.getData()).thenReturn("Mock Data");

MyService service = new MyService(mockApi);

String result = service.fetchData();

assertEquals("Mock Data", result);

}

**Output :**

Running MyServiceTest...

testExternalApi() passed

- Mocked ExternalApi.getData() returned "Mock Data"

- assertEquals: "Mock Data" == fetchData() result

BUILD SUCCESSFUL

Total tests run: 1, Failures: 0, Skipped: 0

Mockito exercises : Exercise 2: Verifying Interactions

import static org.mockito.Mockito.\*;

import org.junit.jupiter.api.Test;

import org.mockito.Mockito;

// Reusing the same ExternalApi and MyService classes

public class MyServiceTest {

@Test

public void testVerifyInteraction() {

ExternalApi mockApi = Mockito.mock(ExternalApi.class);

MyService service = new MyService(mockApi);

service.fetchData(); // This should internally call mockApi.getData()

verify(mockApi).getData(); // Verifies the interaction happened

}

}

Output :

Running MyServiceTest...

testVerifyInteraction() passed

- Verified: ExternalApi.getData() was called once

BUILD SUCCESSFUL

Total tests run: 1, Failures: 0, Skipped: 0